

П.А. КАЧАНОВ, д.т.н., проф. зав. кафедри НТУ "ХПИ"

А.А.ЗУЕВ, ст. преподаватель НТУ "ХПИ"

ИСПОЛЬЗОВАНИЕ МЕТОДОВ РАСЧЕТА ФОНОВОГО ЗАТЕНЕНИЯ ДЛЯ УЛУЧШЕНИЯ КАЧЕСТВА КОМПЬЮТЕРНО-СИНТЕЗИРОВАННЫХ ИЗОБРАЖЕНИЙ

У статті розглянуті різні методи розрахунку затінення для комп'ютерно-синтезованих зображень. Наведено формулу, яка використовується для визначення ступеня затінення елементів зображення. Розглянуто метод розрахунку затінення з використанням віддаленості точок зображення від спостерігача та наведено рекомендації щодо його практичної реалізації.

The article describes the various methods of shading calculation for computer-synthesized images. Formula which used to determine shading of image pixels is shown. The method of shading which used information about distances from the observer to points of the image is proposed, are given recommendations on its implementation.

Постановка проблемы. Для получения высококачественных, приближающихся по качеству к фотореалистичным изображениям, необходимо учитывать рассеянное освещение, падающее на объекты со всех сторон. Любой объект рассеивает падающий свет, что фактически вносит вклад в освещение самого себя и окружающих объектов. Точный учет такого освещения крайне сложен, и его реализация в реальном масштабе времени на данный момент невозможна в полном объеме.

Анализ литературы. Классическим методом расчета затенения является трассировка большого количества лучей из точки x , в различных направлениях и подсчете соотношения количества лучей, не пересекших ни одного объекта до определенного расстояния к общему количеству лучей [1] – это соотношение и будет являться величиной затенения. Однако для проведения вычислений в реальном масштабе времени такой подход неприменим по соображениям производительности, поэтому на практике применяются различные методы аппроксимации [2-3]. Такая аппроксимация позволяет рассчитать затененность по характеру сходную с затененностью получаемой более точными методами, такими как Radiosity [4].

Для расчета аппроксимации затенения применяются следующие методы:

1) Вычисление так называемой «средней нормали» по всем незаблокированным направлениям, которая затем используется в стандартных алгоритмах локального освещения. Такой метод применим только для статической геометрии.

2) Метод, основанный на «размытии» карты удаленности точек от наблюдателя. При достаточно высоком быстродействии, метод обладает суще-

ственным недостатком – в ряде случаев происходит образование широкой окантовки вокруг объектов.

3) Метод Image Space Ambient Occlusion (ISAO), в котором расчет затенения ведется в пространстве изображения. Метод ISAO является наиболее перспективным для использования, так как может быть реализован с использованием ускорителя трехмерной графики и позволяет при относительно высоком быстродействии получать затенение с необходимой степенью достоверности.

Цель статьи – усовершенствование метода расчета и наложения фонового затенения в пространстве экрана с использованием ускорителя трехмерной графики.

Основной раздел. Существующие методы определения взаимного затенения основаны на определении того, какая часть полусферы построенной вокруг данной точки закрыта близкорасположенными объектами.

$$A(x, y) = \frac{1}{2 \cdot \pi} \cdot \int_{S_H^N} V(x, y) \cdot dS, \quad (1)$$

где S_H^N – передняя полусфера, описанная вокруг точки с координатами x, y ; $V(x, y)$ – функция, которая принимает значение 1, когда точка y видна из точки x , в противном случае она принимает значение 0.

Важной особенностью приведенного выражения, является то, что учет затеняющих объектов ведется до определенного расстояния от точки, для которой определяется затенение.

Различные варианты взаимного затенения представлены на рис. 1. Точка на рис. 1 (а) – получает максимальное количество света, а на рис. 2 (б) минимальное.

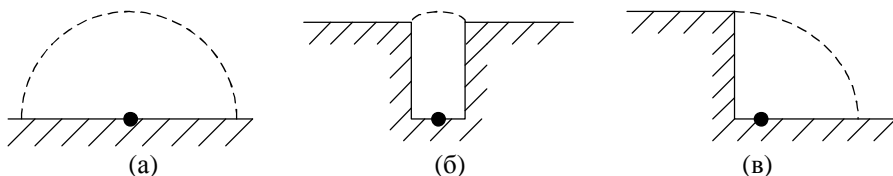


Рис. 1. Варианты взаимного затенения точек

Основные положения метода заключаются в следующем:

- для расчета затенения используются только видимые фрагменты изображения;

- для расчета затенения необходимо знать удаленность каждой точки изображения от наблюдателя – этого достаточно для вычисления затенения;
- затенение рассчитывается относительно положения наблюдателя.

Для определения взаимной видимости точек может использоваться трассировка по полю высот, которое представляет собой расстояния от точек изображения до наблюдателя, но это нецелесообразно по соображениям производительности. Более производительным способом определения видимости является расчет степени загораживания как расстояния Δd на оси наблюдения d , между смещенной в некотором (случайном) направлении точки для которой ведется расчет затенения и позицией соответствующей ей точки изображения (рис. 2).

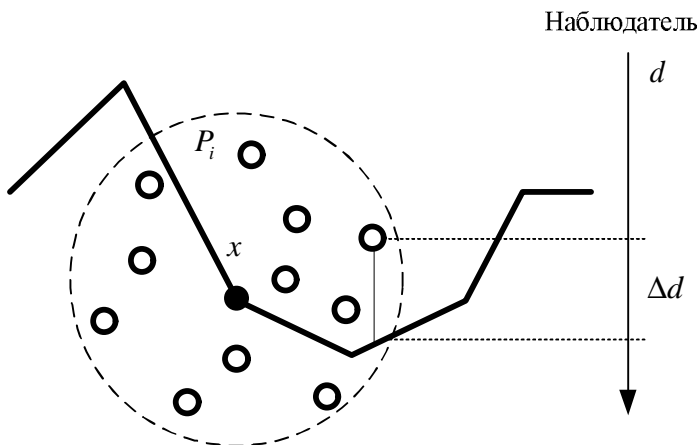


Рис. 2. Точки, используемые для нахождения затенения

Алгоритм расчета затенения состоит из следующих шагов:

- 1) Восстанавливается позиция точки изображения x относительно наблюдателя.
- 2) Создается набор точек P_i , находящихся в окрестности точки x путем ее смещения в случайных направлениях.
- 3) Для точек из множества P_i рассчитываются расстояния Δd на оси наблюдения d до соответствующих им точек изображения.
- 4) Отношение количества точек из множества P_i расположенных ближе к наблюдателю чем соответствующие им точки изображения, к общему количеству точек в множестве и будет являться искомой величиной затенения.

На практике множество P_i содержит ограниченное количество точек, и необходимо обеспечить их индивидуальное случайное распределение для

каждой затеняемой точки изображения, что можно реализовать при помощи выборки смещений из дополнительного двумерного массива псевдослучайных векторов. Выборка должна производиться с некоторым смещением в зависимости от положения затеняемой точки в пространстве экрана.

Функция, определяющая степень затенения точки x

$$O(\Delta d) = \begin{cases} \frac{1}{1 + \Delta d^2}, & \text{при } \Delta d > \epsilon, \\ 0, & \text{при } \Delta d \leq \epsilon \end{cases}, \quad (2)$$

где ϵ - небольшая величина, подбираемая экспериментальным путем.

Все точки, для которых Δd меньше ϵ отбрасываются, так как в этом случае затенения не происходит.

Все операции по расчету затенения могут быть выполнены с использованием ускорителя трехмерной графики, при этом массив удаленностей точек от наблюдателя может быть получен при помощи стандартных методов рендеринга в текстуру. Для хранения псевдослучайных векторов может быть использована двумерная текстура, помещенная в память ускорителя перед началом расчетов. Выборка из нее будет вестись путем смещения текстурных координат.

Выводы. Рассмотренный метод увеличивает точность затенения и его реалистичность, удаляя артефакты аппроксимации. Предложенная модификация не требует сложных расчетов и затрат памяти.

Наложение карты затенения добавляет шум в синтезированное изображение, что несколько снижает качество получаемого изображения. В связи с этим, важной задачей по дальнейшему улучшению методов расчета затенения является удаление шумов на изображении. Для этого полученную карту затенения необходимо фильтровать, при этом фильтрацию необходимо проводить избирательно, для того, чтобы затенение точек фактически расположенных на различных поверхностях не влияло друг на друга.

Список литературы: 1. *Bunnell M.* Dynamic ambient occlusion and indirect lighting / M. Bunnell - In GPU Gems 2. Addison-Wesley, 2005. - pp. 223–233. 2. *Pharr M.* Ambient occlusion / M. Pharr, S. Green - In GPU Gems. Addison-Wesley : 2004. - pp. 279–292. 3. *Shanmugam P.* Hardware accelerated ambient occlusion techniques on GPUs / P. Shanmugam, O. Orkan - In I3D '07: Proceedings of the Symposium on Interactive 3D Graphics, 2007. 4. *Cohen F.* Radiosity and Realistic Image Synthesis / F. Cohen, M. John, W. John - Academic Press, 1993.

Поступила в редакцию 31.05.2010